INT203

Owner's Manual

Weiss Engineering LTD.
Florastrasse 42
8610 Uster
Switzerland

Phone: +41 44 940 20 06
Fax: +41 44 940 22 14
Email: weiss@weiss.ch
Web: www.weiss-highend.com
www.weiss.ch
www.asiaweiss.com

Introduction

Dear customer,

Congratulations on your purchase of the INT203 Firewire I/O Interface and welcome to the family of Weiss equipment owners!

On the following pages I will introduce you to our views on high quality audio processing. These include fundamental digital and analog audio concepts and the INT203 Firewire I/O Interface.

I wish you a long-lasting relationship with your INT203.

Yours sincerely,
Daniel Weiss

President, Weiss Engineering LTD.

Copyright and Liability

Authors: Daniel Weiss & Rolf Anderegg, Weiss Engineering LTD.
TEXLayout: Samuel Groner & Rolf Anderegg, Weiss Engineering LTD.
Date: October 15, 2013

Weiss Engineering LTD. reserves the right to make changes to product specification or documentation without prior notice. Updated manuals and datasheets are available at our website for downloading. Weiss Engineering LTD. makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does part of this manual, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical including photocopying, scanning or any information storage or retrieval system without the express prior written consent of the publisher.

© Copyright Weiss Engineering LTD., 2013

A Short History of Weiss Engineering Ltd.

Having studied electrical engineering, Daniel Weiss joined the Willi Studer (*Studer-Revox*) company in Switzerland. His work included the design of a sampling frequency converter and of digital signal processing electronics for digital audio recorders. In 1985, Mr. Weiss founded the company *Weiss Engineering* LTD. From the outset the company concentrated on the design and manufacture of digital audio equipment for mastering studios. Its first product was the modular 102 Series system. After more than 25 years, this system is still up to date (24 bit/96 kHz).

In the early nineties the Gambit Series was launched, taking ergonomics and sonic quality to new heights. The Gambit Series consists of stand-alone units like equalizer, denoiser/declicker, dynamics processor, A/D converter, D/A converter, sampling frequency converter, dithering etc. 40 bit floating point processors and sampling rates up to 192 kHz are employed. In 2001 we have decided to enter the High-End Hi-Fi market, which offers a comparable clientele to that of the mastering studios. Both consist of critical and discerning listeners, who are in constant search for the best audio reproduction equipment or the best audio tools respectively.

Our list of clients includes big names, like *SONY*, *BMG*, *EMI*, *Warner*, *Hit Factory*, *Abbey Road*, *Teldec*, *Telarc*, *Gateway Mastering* (Bob Ludwig), *Bernie Grundman Mastering*, *Masterdisk*, *Sterling Sound*, *Whitfield Street*, *Metropolis* and hundreds more. For a more comprehensive list you are invited to visit our pro audio website at www.weiss.ch.

Today Weiss Engineering LTD. employs nine people, five of them in the engineering department.

Our Mission and Product Philosophy

The wealth of experience we have gained in over 25 years of designing products for top mastering engineers, we now also apply to the design of outstanding High-End Hi-Fi products. Our mission is to create equipment which becomes classic right from the outset — outstanding in sonics and design. These are some of the milestones at Weiss Engineering LTD.:

1985 Introduction of the 102 Series, a 24 bit modular digital audio processor for mastering studios

1986 Introduction of one of the first sample rate converters for digital audio

1987 Introduction of one of the first digital equalizers

1989 Introduction of one of the first digital dynamics processors

1991 Introduction of the Ibis digital mixing console, built for the mix-down of classical music

1993 Introduction of the Gambit Series of digital audio processors, which employ 40 bit floating point processing and sport an extremely ergonomic user interface

1995 First 96 kHz sampling rate capable products delivered

2001 Introduction of the MEDEA High-End Hi-Fi D/A Converter, the first product in our High-End series

2004 Introduction of the JASON CD Transport

2007 Introduction of the CASTOR High-End Hi-Fi Power Amplifier

2008 Introduction of the MINERVA Firewire D/A Converter and the VESTA Firewire-AES/EBU interface

2010 Introduction of the DAC202 Firewire D/A Converter, the INT202 Firewire Interface and the ATT202 Passive Attenuator

2011 Introduction of the MAN301 Music Archive Network Player and the INT203 Firewire I/O Interface

2012 Introduction of the MEDEA+ D/A Converter, the MEDEA with an enhanced analog section

2013 Introduction of the INT204 USB/DSD Interface and the MEDUS State Of The Art D/A Converter. Continuing development of the MAN301 Network Player with new software features.

Contents

Introduction iii

A Short History of Weiss Engineering Ltd. v

1 Advanced Digital & Analog Audio Concepts Explained 3
- 1.1 Jitter Suppression & Clocking . . . 3
- 1.2 Jitter handling in the INT203 Firewire I/O Interface . . . 4
- 1.3 Up-/Oversampling & Sampling Rate Conversion in General . . . 5
- 1.4 Dithering . . . 7
- 1.5 Digital Level Control . . . 8
- 1.6 Firewire vs. USB . . . 10

2 The INT203 Firewire I/O Interface 13
- 2.1 Features . . . 13
  - 2.1.1 Firewire . . . 13
  - 2.1.2 Outputs . . . 13
  - 2.1.3 Inputs . . . 13
  - 2.1.4 Synchronization . . . 14
  - 2.1.5 Power Supply . . . 14
  - 2.1.6 Stand-Alone Operation as a Digital Level Control Device 15
  - 2.1.7 Firmware features . . . 15
  - 2.1.8 Frontpanel . . . 16
  - 2.1.9 Backpanel . . . 16
- 2.2 Installation . . . 16
  - 2.2.1 Software installation . . . 16
- 2.3 Operation . . . 19
  - 2.3.1 Firewire and Power Connection . . . 19
  - 2.3.2 Output via AES/EBU or S/PDIF Connection . . . 19
  - 2.3.3 Input via AES/EBU and/or S/PDIF Connection . . . 20
  - 2.3.4 Bit Transparency Check . . . 20
  - 2.3.5 Remote Control . . . 21
  - 2.3.6 Stand-Alone Operation for Level Control . . . 22
  - 2.3.7 Software Setup . . . 22

3 INT203 Technical Data 25

3.1 Digital Inputs 25
3.2 Digital Outputs 25
3.3 Synchronization 25
3.4 Power 26
3.5 Level Control 26
3.6 Bit Transparency Check 26
3.7 Polarity Control 26

A Contact **27**

Chapter 1

Advanced Digital and Analog Audio Concepts Explained

1.1 Jitter Suppression and Clocking

What is jitter and how does it affect audio quality? In the audio field the term jitter designates a timing uncertainty of digital clock signals. In an analog to digital converter (A/D) the analog signal is sampled (measured) at regular time intervals; in the case of a CD, 44 100 times a second or every 22.675737 μs (microseconds). If these time intervals are not strictly constant then one talks of a jittery conversion clock. In practice it is of course not possible to generate *exactly* the same time interval between each and every sample. After all, even digital signals are analog in their properties and thus are influenced by noise, crosstalk, power supply fluctuations, temperature etc.

Hence a jittery clock introduces errors to the measurements taken by the A/D, resulting from measurements being taken at the wrong time. One can easily observe that the level of the error introduced is higher during high audio frequencies, because high frequency signals have a steeper signal form. A good designer takes care that the jitter amount in his/her design is minimized as well as possible.

What type of equipment can be compromised by jitter? There are three types: The A/D converter as described above, then there is the D/A converter where the same mechanism as in the A/D converter applies and the third is the asynchronous sample rate converter (ASRC). The ASRC is not something usually found in Hi-Fi systems. It is used by sound engineers to change the sample rate from e.g. 96 kHz to 44.1 kHz, or e.g. for putting a 96 kHz recording onto a 44.1 kHz CD.

You may now argue that in High-End Hi-Fi there are such things as „oversamplers“ or „upsamplers“. Yes, those are in essence sampling rate converters,

however in a well-designed system these converters employ a synchronous design, where jitter does not play any role. Of course a conversion between 96 kHz and 44.1 kHz as in the example above can be done in a synchronous manner as well. An ASRC in fact is only required either where one or both of the sampling frequencies involved are changing over time („varispeed“ mode of digital audio recorders) or where it is unpractical to synchronize the two sampling frequencies.

So basically in Hi-Fi jitter matters where there are A/D or D/A converters involved. CD and DVD players are by far the most numerous type of equipment employing D/A converters. And of course stand-alone D/A converters. Jitter, being an analog quantity, can creep in at various places. The D/A converter built into CD or DVD players can be „infected“ by jitter through various crosstalk mechanisms, like power supply contamination by power hungry motors (spindle/servo), microphony of the crystal generating the sampling clock or capacitive/inductive crosstalk between clock signals etc.

In the standalone D/A converter jitter can be introduced by inferior cables between the source (e.g. CD transport) and the D/A converter unit or by the same mechanisms as described above — except for the motors of course. In the case of a stand-alone D/A converter (as the DAC202), one has to take two different jitter contamination paths into account. One is the internal path where internal signals can affect the jitter amount of the sampling clock generator. Here, all the good old analog design principles have to be applied. Such as shielding from electric or magnetic fields, good grounding, good power supply decoupling, good signal transmission between the clock generator and the actual D/A chip.

The other path is the external signal coming from the source to which the sampling clock has to be locked. In other words the D/A converter has to run synchronous to the incoming digital audio signal and thus the frequency of the internal sampling clock generator has to be controlled so that it runs at the same sampling speed as the source (e.g. CD transport). This controlling is done by a phase locked loop (PLL), which is a control system with error feedback. Of course the PLL has to be able to follow the long term fluctuations of the source, e.g. the sampling rate of the source will alter slightly over time or over temperature, it will not be a constant 44.1 kHz in the case of a CD. But the PLL should not follow the short-term fluctuations (jitter). Think of the PLL as being like a very slow-reacting flywheel.

1.2 Jitter handling in the INT203 Firewire I/O Interface in more detail

The Jitter Elimination Technologies (JET) PLL on the chip used in the INT203 feature state-of-the art jitter rejection abilities and extremely low intrinsic jitter levels. Like all phase-locked loops, JET PLL use feedback to lock an oscillator to a timing reference. They track slow reference changes, but effectively free-run through rapid modulations of the reference (i.e. flywheel like). From a jitter

transfer point of view, they provide increasing jitter attenuation above some chosen corner frequency. Jitter attenuation is just one aspect of PLL design. Other considerations include frequency range and intrinsic jitter. It can be shown that conventional designs are bound by a fundamental tradeoff between these three aspects. For example, specifying a frequency range of one octave means using a low-Q oscillator. But that makes for high intrinsic jitter when the loop corner frequency is held down. Conversely, good jitter attenuation and low intrinsic jitter can be had by using a voltage-controlled crystal oscillator (VCXO). But the frequency range is then tiny. A further consideration is that only low-Q oscillators are easy to integrate on chip. JET PLL sidestep the above-mentioned tradeoff. It incorporates two loops. One is largely or wholly numeric, and has its corner frequency set low enough to give good reference-jitter attenuation. The other regulates the analog oscillator and has its corner frequency set much higher, to moderate the intrinsic jitter. The two corner frequencies might be around 10 Hz and 100 kHz, for example. Another benefit of having a high corner frequency in the analog loop is that interference, e.g. via the oscillator's supply rail, is more-effectively suppressed. JET PLL requires a fast, stable, fixedfrequency clock. It is this that gives it stability in the band between the two corner frequencies. (Equally, in this band any jitter on this clock passes straight through to the JET PLL's clock output.) The stable clock is usually derived from a free-running crystal oscillator. JET PLL contains a number-controlled oscillator, which can also be called a fractional frequency divider. Like the analog oscillator, this injects jitter. Typically, spectrum shaping is used to push most of that jitter up to frequencies where it will be heavily attenuated by the analog loop. As well as frequency-locking the analog oscillator to the provided reference, JET PLL can also phase-lock an associated frame sync to the reference.

1.3 Upsampling, Oversampling and Sampling Rate Conversion in General

In consumer audio circles the two terms oversampling and upsampling are in common use. Both terms essentially mean the same, a change in the sampling frequency to higher values. Upsampling usually means the change in sampling rate using a dedicated algorithm (e.g. implemented on a digital signal processor chip (DSP)) ahead of the final D/A conversion (the D/A chip), while oversampling means the change in sampling rate employed in today's modern D/A converter chips themselves.

But let's start at the beginning. What is the sampling frequency? For any digital storage or transmission it is necessary to have time discrete samples of the signal which has to be processed. I.e. the analog signal has to be sampled at discrete time intervals and later converted to digital numbers (also see section *Jitter Suppression and Clocking*, p. 3). This sampling and conversion process

happens in the so-called analog to digital converter (A/D). The inverse in the digital to analog converter (D/A).

A physical law states that in order to represent any given analog signal in the digital domain, one has to sample that signal with at least twice the frequency of the highest frequency contained in the analog signal. If this law is violated so called aliasing components are generated which are perceived as a very nasty kind of distortion. So if one defines the audio band of interest to lie between 20 Hz and 20 kHz, then the minimum sampling frequency for such signals must be 40 kHz.

For practical reasons explained below, the sampling frequency of 44.1 kHz was chosen for the CD. A sampling frequency of 44.1 kHz allows to represent signals up to 22.05 kHz. The designer of the system has to take care that any frequencies above 22.05 kHz are sufficiently suppressed before sampling at 44.1 kHz. This suppression is done with the help of a low pass filter, which cuts off the frequencies above 22.05 kHz. In practice such a filter has a limited steepness, i.e. if it suppresses frequencies above 22.05 kHz it also suppresses frequencies between 20 kHz and 22.05 kHz to some extent. So in order to have a filter which sufficiently suppresses frequencies above 22.05 kHz, one has to allow it to have a so-called transition band between 20 kHz and 22.05 kHz where it gradually builds up its suppression.

Note that so far we have talked about the so-called anti-aliasing filter, which filters the audio signal ahead of the A/D conversion process. For the D/A conversion, which is of more interest to the High-End Hi-Fi enthusiast, essentially the same filter is required. This is because after the D/A conversion we have a time discrete analog signal, i.e. a signal that looks like steps, having the rate of the sampling frequency.

Such a signal contains not only the original audio signal between 20 Hz and 20 kHz but also replicas of the same signal symmetrical around multiples of the sampling frequency. This may sound complicated, but the essence is that there are now signals above 22.05 kHz. These signals come from the sampling process. There are now frequencies above 22.05 kHz which have to be suppressed, so that they do not cause any intermodulation distortion in the amplifiers and speakers, do not burn tweeters or do not make the dog go mad.

Again, a low pass filter, which is called a „reconstruction filter“, is here to suppress those frequencies. The same applies to the reconstruction filter as to the anti-aliasing filter: pass-band up to 20 kHz, transition-band between 20 kHz and 22.05 kHz, stop-band above 22.05 kHz. You may think that such a filter is rather „steep“, e.g. frequencies up to 20 kHz go through unaffected and frequencies above 22.05 kHz are suppressed to maybe $\frac{1}{100\,000}^{\text{th}}$ of their initial value. You are right, such a filter is very steep and as such has some nasty side effects. For instance it does strange things to the phase near the cutoff frequency (20 kHz) or it shows ringing due to the high steepness. In the early days of digital audio these side effects have been recognized as being one of the main culprits for digital audio to sound bad.

So engineers looked for ways to enhance those filters. They can't be eliminated because we are talking laws of physics here. But what if we run the whole thing at higher sampling rates? Like 96 kHz or so? With 96 kHz we can allow frequencies up to 48 kHz, so the reconstruction filter can have a transition band between 20 kHz and 48 kHz, a very much relaxed frequency response indeed. So let's run the whole at 96 kHz or even higher! Well — the CD stays at 44.1 kHz. So in order to have that analog lowpass filter (the reconstruction filter) to run at a relaxed frequency response we have to change the sampling frequency before the D/A process. Here is where the upsampler comes in. It takes the 44.1 kHz from the CD and upsamples it to 88.2 kHz or 176.4 kHz or even higher. The output of the upsampler is then fed to the D/A converters, which in turn feeds the reconstruction filter. All modern audio D/A converter chips have such an upsampler (or oversampler) already built into the chip. One particular chip, for instance, upsamples the signal by a factor of eight, i.e. 44.1 kHz ends up at 352.8 kHz. Such a high sampling frequency relaxes the job of the reconstruction filter very much; it can be built with a simple 3^{rd} order filter.

So, how come that upsamplers are such a big thing in High-End Hi-Fi circles? The problem with the upsamplers is that they are filters again, digital ones, but still filters. So in essence the problem of the analog reconstruction filter has been transferred to the digital domain into the upsampler filters. The big advantage when doing it in the digital domain is that it can be done with a linear phase response, which means that there are no strange phase shifts near 20 kHz and the ringing can also be controlled to some extent. Digital filters in turn have other problems and of course have quite a few degrees of freedom for the designer to specify. This means that the quality of digital filters can vary at least as much as the quality of analog filters can. So for a High-End Hi-Fi designer it is a question whether the oversampling filter built into the D/A chips lives up to his/her expectations. If not, he/she can choose to design his/her own upsampler and bypass part of or the whole oversampler in the D/A chip. This gives the High-End Hi-Fi designer yet another degree of freedom to optimize the sonic quality of the product.

1.4 Dithering

You have probably not heard the term dithering in conjunction with audio. Actually it is a term widely used in the professional audio realm but not so much in the High-End Hi-Fi market.

What is dithering? Suppose a digital recording has been made with a 24 bit A/D converter and a 24 bit recorder. Now this recording should be transferred to a CD, which has just 16 bits per sample, as you know. What to do with those 8 bits, which are too many? The simplest way is to cut them off, truncate them. This, unfortunately, generates harmonic distortions at low levels, but which nonetheless cause the audio to sound harsh and unpleasant. The harmonic distortion is generated because the eight bits, which are cut off from the 24 bits,

are correlated with the audio signal, hence the resulting error is also correlated and thus there are distortions and not just noise (noise would be uncorrelated). The dithering technique now is used to de-correlate the error from the signal. This can be achieved by adding a very low level noise to the original 24 bit signal before truncation. After truncation the signal does not show any distortion components but a slightly increased noise floor. This works like magic — the distortion is replaced by a small noise which is much more pleasant. I have given the example of a 24 bit recording, which has to be truncated to 16 bits. Where is the application in High-End Hi-Fi audio? More and more signal processing is implemented in the digital domain. Think of digital equalizers, digital volume controls, upsamplers, digital pre-amplifiers, decoders for encoded signals on DVD etc. All those applications perform some mathematical operations on the digital audio signal. This in turn causes the wordlength of the signal to be increased; an input signal to an upsampler may have a wordlength of 16 bits (off a CD), but the output signal of the upsampler may have 24 bits or even more. This comes from the fact that the mathematical operations employed in such devices increase the word length. A multiplication of two 2-digit numbers results in a four-digit number. So after the upsampler the word length may be higher than the subsequent processor may be able to accept. In this example, after the upsampler there may be a D/A converter with a 24 bit input word length capability. So if the upsampler generates a word length of more than 24 bits it should be dithered to 24 bits for maximum signal fidelity.

Another application where dither is important is described in the next paragraph.

1.5 Digital Level Control

In High-End Hi-Fi circles a level control done in the digital domain is often viewed as being inferior to one operating in the analog domain. Let's look on how a digital level control works and why it can be an excellent solution if it is properly implemented.

A level control is a multiplication of the audio signal with a constant, the ,,gain factor". The gain factor usually is in the range of zero (signal fully off) to one (signal untouched). A factor of 0.5 then means that the audio signal is attenuated to half of its amplitude. What exactly happens when we multiply two numbers? If we e.g. multiply a 2 digit and a 3 digit number, the resulting number can be up to 5 digits long (the sum 2 plus 3). As an example: 30 times 500 equals 15 000; 2 digits times 3 digits yields a 5 digit result.

In digital audio, the numbers are represented in the binary system, not the decimal system. A decimal number consists of digits 0 through 9, a binary number of digits 0 and 1. So a binary number may look like this: 1011 0011 0101 1101. This is a 16 digit or 16 bit binary number, the grouping into 4 bit chunks is for better readability. The audio samples on a CD are represented with such a binary number system with each sample value represented with 16 bits. Now

let's assume we have a 8 bit gain factor for a level control. If we apply that to a signal coming off a CD we multiply a 8 bit gain factor with a 16 bit sample value. The result is up to 24 bits long (the sum of the wordlengths of the two factors). An example:

$$0100\ 1001 \times 1001\ 0110\ 0111\ 1011 = 0010\ 1010\ 1110\ 1001\ 0001\ 0011$$

The question now is what do we do with the 24 bit long result? The digital to analog converter which converts the samples after the level control may only be capable to handle 16 bit wide samples. Thus what should we do with the excessive 8 bits? The simplest solution is to truncate the 24 bit sample to 16 bits, i.e. to cut off the 8 least significant bits. The truncated 24 bit result above then would look like this: 0010 1010 1110 1001 i.e. the first 16 bits of the 24 bit result above.

The remaining bits (0001 0011) are discarded. If these bits are discarded an error is introduced. This error is called a quantization error, because the 24 bit result is requantized to 16 bits. Unfortunately the quantization error is part of the audio signal — and if we take that part away from the signal, the signal undergoes some distortion, the so called quantization distortion. The sound example at the link below shows how such a distortion sounds. In this music example a 16 bit signal is truncated to 8 bits; 8 bits in order to clearly show the effect. Notice how the noise (distortion) is modulated by the music signal.

www.weiss-highend.ch/computerplayback/nodither.mp3

This is how a badly implemented digital level control works... Fortunately there is a better way to handle the re-quantizing. One solution would be to use a D/A converter with a higher wordlength, e.g. a 24 bit converter, to accommodate for the 24 bit samples coming out of the level control. This of course would already help a lot, but there is another technique: dithering.

The idea about dithering is to de-correlate the quantization error from the audio signal. As we have seen in the example above, the quantization error depends on the audio signal (i.e. it is correlated with the audio signal). On the other hand, if dither noise is added to the 24 bit sample after the level control and before the re-quantization to 16 bits, the quantization error can be fully decorrelated from the signal. This means instead of distortion there is noise. The music is undistorted. The audio example at the link below is again a 16 bit signal quantized to 8 bits, but with dither noise added. A much more pleasant experience. Notice how the noise stays untouched by the music, i.e. there is no noise modulation.

www.weiss-highend.ch/computerplayback/flatdither.mp3

Dithering does not stop here. More elaborate dithering schemes shape the noise such that it is mainly present at higher frequencies where the human ear is less sensitive. This means that the audible noise is much lower. The link below is again the 16 bit source quantized to 8 bits with noise-shaped dithering. Probably hard to believe that this is only an 8 bit system! Note that the music is not

distorted at all, despite the 8 bit resolution. Remember, a 16 bit system has 65 536 quantization steps while a 8 bit system has only 256 quantization steps — a huge difference. And still, the properly dithered 8 bit system sounds great.

www.weiss-highend.ch/computerplayback/shapeddither.mp3

This is what a properly dithered level control is capable to do. You have heard the 8 bit version, imagine that with today's 24 bit converters — no question that a level control with a 24 bit wordlength easily rivals the best analog level controls. By the way, 24 bits means 16 777 216 quantization steps.

The last example below toggles the noiseshaping dither on and off to give a good contrast between dither/no dither versions.

www.weiss-highend.ch/computerplayback/togglingdither.mp3

Dithering is used in many disciplines. Also see

en.wikipedia.org/wiki/Dithering

A paper on this topic (including image examples) can be found here:

www.weiss-highend.ch/computerplayback/Digital_Level_Control.pdf

I hope these excursions into the theory and practice of audio engineering has been useful for you. If you would like to dive further into those issues I recommend visiting the website of Mr. Bob Katz, a renowned mastering engineer and a Weiss Engineering LTD. customer. He publishes articles on dithering and jitter and many other topics at

www.digido.com

1.6 Firewire vs. USB

Firewire is a peer-to-peer protocol, meaning that every device on a Firewire network is equally capable of talking to every other device. Two video cameras on a Firewire network can share data with each other. A Firewire audio interface could save sound data directly to a Firewire hard drive. Your computer is just another peer on this network, and has no inherent special status. Firewire is always implemented in hardware, with a special controller chip on every device. So the load it puts on your CPU is much lighter than USB communications load, and you're much less likely to lose any sound data just because you're running fifteen things at once. Specialized hardware usually makes things faster and more reliable, and this is one of those times. But the real reason Firewire is more reliable than USB is more fundamental than that. It's because Firewire allows two operating modes. One is asynchronous, similar to what USB uses. The other is isochronous mode, and it lets a device carve out a certain dedicated amount of bandwidth that other devices cant touch. It gets a certain number of time slices each second all its own. The advantages for audio should be obvious: that stream of data can just keep on flowing, and as long as there isn't more bandwidth demand than the wire can handle (not very likely) nothing will interfere with it.

No collisions, no glitches. From a practical perspective, this also makes it safer to send a lot more audio via Firewire. That's why most of the multichannel interfaces (18 channels, 24 channels, etc.) are Firewire devices, and USB devices usually just send a two channel stereo signal.

For hooking up your mouse, keyboard or thumb drive, USB is plenty fast and plenty cheap. For hard drives, either one will do (although Firewire is somewhat more reliable). For audio devices, USB will do fine if no other devices are competing with it and if you have processor room to spare. But Firewire will always be able to handle more load with lower latency and no glitches, because it has resources it can set aside to make sure your audio gets where it needs to go.

Chapter 2

The INT203 Firewire I/O Interface

2.1 Features

The INT203 Firewire I/O Interface extends your computer with digital audio inputs and outputs. It allows you to:

- playback audio from your computer to digital audio equipment (D/A converter, digital audio processing, etc.)
- connect your computer to digital audio sources (A/D converter, digital audio transports, etc.) in order to record/route digital audio
- use the INT203 as a stand-alone digital level control when not connected to a computer

2.1.1 Firewire

There are two Firewire connectors in order to chain several Firewire based units on a single Firewire bus. Which of the two Firewire connectors is used for the connection to the computer is not relevant. The supported sampling frequencies are 44.1, 48, 88.2, 96, 176.4 or 192 kHz.

2.1.2 Outputs

There are two AES/EBU outputs on a XLR and a RCA connector available on the backpanel (fig. 2.3). The XLR and RCA connectors are fed from the same source (Firewire output) in single wire mode, but have separate output stages. They can be used simultaneously.

2.1.3 Inputs

Two separate digital audio inputs, one XLR connector and one RCA connector, are available in order to transmit digital audio via Firewire into the connected

computer. Thus two separate digital stereo sources will be available (for recording, routing, etc.) within the computer's audio system (4 channels). Since certain recording applications simply call for stereo sources, the primary input switch (fig. 2.3) allows you to select which of the input connectors is chosen as the primary of the two stereo inputs.

2.1.4 Synchronization

The synchronization source and rate of the INT203 can be freely chosen among:

Internal an internal clock is generated within the INT203. Typically chosen when the INT203 is used as a digital output for connecting to external equipment. With the proper (advanced) player software the sampling rate of the INT203 is switched automatically depending on the track played. This is a key issue in order to achieve bit transparency (see also section 2.3.4).

AES/EBU XLR synchronization is achieved by an external source connected via the XLR input. Typically chosen when a digital audio source is connected (via XLR) in order to transfer audio to the computer (e.g. for recording).

S/PDIF RCA synchronization is achieved by an external source connected via the RCA input. Typically chosen when a digital audio source is connected (via RCA) in order to transfer audio to the computer (e.g. for recording).

In all cases the INT203 will function as the synchronization master for the computer. The INT203's intrinsic jitter is very low. When synchronizing to external sources the INT203 will automatically detect the sampling rate and sync-lock to it. Synchronization settings are controlled via the *Weiss Firewire IO* window. On a Mac system the sampling rate set in *AudioMidi* sets the one in the *Weiss Firewire IO* window and vice versa.

> *Since certain audio applications on your computer (e.g. recording tools) rely on the sync rate setting within the* Weiss Firewire IO *control panel, make sure the synchronization settings are set to the correct rate before recording or when playing back audio.*

2.1.5 Power Supply

The power can be either supplied from the Firewire bus or from the external power supply supplied with the INT203.

Firewire bus power may or may not be available with the particular computer used. Refer to the computer manual to find out. If the Firewire connector at the computer is a 4 pin type (fig. 2.1) then bus power is not available by definition. With other types of connectors (6 pin or 9 pin) bus power may be available.

If the INT203 powers up with the computer connected and with no external power supply connected, then obviously bus power is available. If preferred, the external power supply may still be connected in that case. The INT203 then automatically switches over to the external power supply.

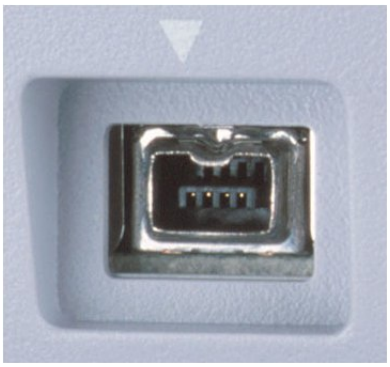

Figure 2.1: 4 pin Firewire socket

2.1.6 Stand-Alone Operation as a Digital Level Control Device

If the INT203 is not connected to a computer (no Firewire connection), the INT203 enters the stand-alone level control mode. In this mode the input selected with the input selector switch (RCA or XLR connector) is fed to both outputs (RCA and XLR) via the level control algorithm. With the (optional) IR remote control unit, the level of the digital signal can be changed. The INT203 firmware features as described below are still supported in the level control mode. Namely the bit transparency check and the absolute polarity reversal switch. Note, if the INT203 is not connected via Firewire, an external power supply needs to be connected to the INT203.

2.1.7 Firmware features

The firmware in the INT203 allows for level control and for absolute polarity reversal of the digital audio output. These features are only accessible via the (optional) *Remote Control* unit (section 2.3.5). If the remote control unit is not used, the gain stays at 0 dB as set in the factory settings. 0 dB means a fully bit transparent transfer. The absolute polarity is set to 0°, i.e. the polarity is not changed. The level control is properly dithered, i.e. there are no artifacts due to the level control. Refer to previous sections for more background information on dithering (1.4) and digital level control (1.5).

In addition the firmware allows for testing bit transparency of the music player program running on the computer. This is useful to find out the proper settings/configuration of the player program to achieve bit transparency at the

INT203's output. Bit transparency means that the bits fed from the harddisk to the INT203 are not changed in any manner.

2.1.8 Frontpanel

- *Power* switch
- *Power/Bit Transparency Check* LED
- *Bit Transparency Check* switch
- Infrared receiver (for optional remove control)
- 2 6-pin Firewire 400 sockets, which one is used does not matter
- Power socket for an external power supply

2.1.9 Backpanel

- *IN1/IN2* primary input select switch
- 1 XLR output connector: *OUT1* stereo signal output (single wire mode)
- 1 RCA output connector: *OUT2* carries the same stereo signal output as *OUT1*
- 1 XLR input connector: *IN1* stereo signal input (single wire mode)
- 1 RCA input connector: *IN2* stereo signal input (single wire mode)

2.2 Installation

Carefully unpack the INT203. The following items should be enclosed:

- the INT203 Firewire I/O Interface unit
- a Power Supply
- a CD with the necessary Firewire drivers for Windows and OSX and with the Bit Transparency check files.
- this Owner's Manual
- a Certificate of Guarantee
- optionally: an IR Remote Control

2.2.1 Software installation

Perform the following installation procedure before connecting the INT203 to the computer. The necessary files are supplied on the enclosed drivers CD.

Figure 2.2: INT203 frontpanel

Figure 2.3: INT203 backpanel

Windows:

1. Do not connect the device.
2. Double click *WeissFirewireInstaller.exe*
3. Click *Next*
4. Select the directory where you'd like to install the tools. Usually you can use the default values and click *Next*
5. Select if you'd like to create a desktop icon. *Next*
6. Click *Install*
7. You will be asked if you'd like to continue the installation because the driver/software didn't pass the Windows-Logo-Test. Select *Continue.*
8. Select *Yes, restart the computer now* and click *Finish*

Mac:

1. Do not connect the device.
2. Mount the WeissFirewire.dmg diskimage by double clicking it
3. From the mounted drive double click *WeissFirewire-3.5.5.10185.pkg* (obviously this version number may increase over time)
4. Click *Continue*
5. Select the drive (usually you leave it at the defaults)
6. Click *Continue*
7. Click *Upgrade* or *Install*
8. You'll be asked to login as administrator
9. Confirm *Continue Installation* when warned that the computer requires a reboot after install.
10. Click Restart

After installation of the software connect the INT203 to the computer.

> *Make sure to plug in the Firewire cable in the proper orientation! If the connector is plugged in reversed the INT203 can be destroyed because of the bus power on the Firewire bus.*

The power LED at the INT203 may be lit. If this is the case then the INT203 is powered from bus power on the Firewire bus. If the LED is not lit then an

appropriate external power supply (available from Weiss) has to be connected to the INT203 and the power switch at the INT203 has to be switched to *on*. The INT203 should now be recognized automatically. In Windows tell the installation window that you do not want to check the Microsoft website for drivers and then let the drivers be installed automatically. Ignore warnings concerning *Windows Logo Test* and continue the installation until completed. You will be asked to install drivers for *Weiss Engineering Ltd. –Firewire Unit–*.

2.3 Operation

2.3.1 Firewire and Power Connection

First install the software on your computer as described in section 2.2.1. Then connect the computer to the INT203 via an appropriate Firewire cable. There are three types of cables applicable:

- 4 pin Firewire 400 (computer side) to 6 pin Firewire 400 (INT203 side)
- 6 pin Firewire 400 (computer side) to 6 pin Firewire 400 (INT203 side)
- 9 pin Firewire 800 (computer side) to 6 pin Firewire 400 (INT203 side)

Choose the proper cable as required by your computer.

> *Make sure to plug in the Firewire cable in the proper orientation! If the connector is plugged in reversed the INT203 can be destroyed because of the bus power on the Firewire bus.*

After connecting the INT203 to the computer and switching the computer on and switching the INT203 power switch to *on*, the power LED at the INT203 may be lit. If this is the case then the INT203 is powered from bus power on the Firewire bus. If you connect an external power supply to the INT203 then the Firewire bus power is automatically switched off and the INT203 is powered from the external power supply.

2.3.2 Output via AES/EBU or S/PDIF Connection

Connect the output of the INT203 to your D/A Converter or any other digital audio device. Use either the XLR or RCA connectors as applicable. The INT203 does *not* support output in *dual wire* format which is used by some D/A converters for higher sampling rates. Refer to Weiss' INT202 (*weiss-highend.ch/int202*) product which supports *dual wire*.

Connect any of the two outputs of the INT203 to the input of your DAC. Both outputs can be used simultaneously, i.e. you may connect more than one DAC at the same time.

2.3.3 Input via AES/EBU and/or S/PDIF Connection

Connect the output of any digital audio source to the INT203. Use either the XLR or RCA connectors as applicable. Make sure that the INT203 is synchronized to the connected source, resp. sources. Choose the correct master sync source as applicable in your digital audio setup (refer to section 2.1.4). Automatic sync lock is attempted when an external sync source is selected. Both audio inputs (XLR and RCA) transmitted via Firewire and are now available within your computer's audio system at the given sync sampling rate. Certain applications do not support audio input with more than two channels (stereo), in which case you may need to apply the primary input switch in order to access your desired input signal.

> *Note that any change of the primary input selection switch (fig. 2.3) setting takes effect only after the INT203 has been switched off (Power LED not lit) and switched on again (Power LED lit).*

2.3.4 Bit Transparency Check

If the bit transparency check switch (*TRSP CHECK* switch on the frontpanel, see fig. 2.2) is on, the INT203 mutes its outputs and enters the bit transparency check mode. It is looking for specific bit patterns coming from the Firewire source. I.e. in order to test the bit transparency of the player software on the computer the enclosed WAV files (see driver CD or download from *www.weiss.ch/downloads/dac202fw/dac202-trnz-wav.zip*) have to be ripped to the computer and played from the player program. When the INT203 detects the expected bits coming from such a file, it flashes the power LED continuously, indicating that the player is bit transparent. If the LED stays lit the player is not bit transparent. There are a total of 12 files supplied. One set for 16 bit and one set for 24 bit transparency checking. One set includes all supported sampling rates, namely 44.1, 48.0, 88.2, 96.0, 176.4, 192.0 kHz. This allows to make sure the player is bit transparent for all sampling rates. We suggest to check first with the 24 bit files. If those yield a bit transparent result then it is not necessary to check with the 16 bit files.

If the player does not seem to be bit transparent then this can have several causes, such as:

- a volume control not at 0 dB gain
- an equalizer
- a sampling rate conversion
- a *sound enhancer* feature and more

Make sure all those processing elements are bypassed. Particularly the sampling rate conversion can creep in *unnoticed*. I.e. the sampling rate in the *Weiss Firewire IO* window has to match the sampling rate of the file played, else a conversion is going on in the operating system. For iTunes there is another issue to consider: Whenever the sampling rate is changed in the *AudioMidi* setup or the *Weiss Firewire IO* window, the iTunes program has to be restarted to regain bit transparency. For iTunes running on a Mac computer a program like Sonic Studio's *Amarra* is highly recommended. *Amarra* switches the sampling rate in *AudioMidi* (i.e. in the INT203) automatically depending on the sampling rate of the file played. *Amarra* works in conjunction with iTunes. On a Windows based system the use of *ASIO* or *WASAPI* is highly recommended. These systems make it simple to achieve bit transparent playback. In addition the sampling rate of the INT203 is switched automatically depending on the sampling rate of the file played.

2.3.5 Remote Control

If you have acquired the remote control (fig. 2.4) for the INT203 then you can control the volume and the polarity of the digital output signal transferred by the INT203. The volume up/down switches (△/▽) control the volume accordingly.

Figure 2.4: INT203 remote control (optional)

The maximum gain is 0 dB (a gain factor of 1.0), i.e. the INT203 can not amplify the signal and thus makes sure that there are no *overs* occurring within the INT203. The *mute* switch allows to toggle between fully off (muted) and the volume set with the volume control. Absolute Polarity control: Pressing the mute switch for a few seconds toggles the operation mode to polarity control. In that mode the △ key switches the polarity to 0° and the ▽ key switches to 180° (i.e. inverted polarity). Press the mute key for a few seconds again to return to the volume control mode. The polarity set before is retained when switching back to volume control mode. When changing the polarity with the △ and ▽ keys you notice a short muting of the audio. This prevents any clicks caused by the polarity switching and is also an indication that the switching has been applied.

The volume setting is retained when switching the INT203 off/on, while the polarity is reset to not inverted after a power cycle.

2.3.6 Stand-Alone Operation for Level Control

If the INT203 is not connected to a computer (no Firewire connection), the INT203 enters the stand-alone level control mode. In this mode the input selected with the input selector switch (RCA or XLR connector) is fed to both outputs (RCA and XLR) via the level control algorithm. The INT203 does NOT need to be power-cycled in order to have the input selector switch taking effect. With the (optional) IR remote control unit, the level of the digital signal can be changed. The INT203 firmware features as described above are still supported in the level control mode. Namely the bit transparency check and the absolute polarity reversal switch. Note, if the INT203 is not connected via Firewire, an external power supply needs to be connected to the INT203.

2.3.7 Software Setup

The connected INT203 device can be controlled through the *Weiss Firewire IO* control panel.

Windows

The control panel can be accessed by clicking on the *Weiss Firewire IO* icon on the desktop. Available tabs:

Global Settings/Bus

Master indicates the sync master on the virtual bus in case multiple devices (e.g. INT203s) are connected.

Sampling Rate selects the sampling rate of the device's internal clock generator. When clocked/syncing to one of the input interfaces (XLR, RCA)

the indicated sampling rate does not necessarily reflect the one fed from the external device. In this case set the sampling rate as indicated by the *Measured Sampling Rate* within *Device Settings/General*.

Sync Source selects the clock to which the INT203 should sync to. For playback this is usually the INT203's *Internal* clock generator.

Buffer Size larger buffer sizes increase robustness against dropouts; lower buffer sizes provide low latency.

Operation Mode determines the stability of the system. Try other modes if clicks are encountered during playback.

Global Settings/WDM Enables the WDM driver and allows you to modify the WDM input and output channel routing. In terms of input channel selection, this is an alternative to the primary input selection switch on the INT203s backpanel.

Global Settings/DPC Determines your computer's performance and recommends an operation mode.

Global Settings/System Various utilities to determine the chipset in your computer and to get information on the supported chipset. Required for debugging if problems with the Firewire connection are encountered.

Global Settings/Info Information about the driver version.

Device Settings/General The device settings should be pretty self-explanatory.

Device Settings/Firmware Loader Allows uploading new firmware to the INT203. Not used for normal operation.

Mac OS X

Configure the INT203 via the *Audio MIDI Setup* (*Applications* → *Utilities*) and the *WeissFirewire Control Panel* (*Applications*). Note that most settings controllable from the control panel are available only in Firewire mode.

Global Settings/Bus

Master indicates the sync master on the virtual bus in case multiple devices (e.g. INT203s) are connected.

Sampling Rate selects the sampling rate of the device's internal clock generator. When clocked/syncing to one of the input interfaces (XLR, RCA) the indicated sampling rate does not necessarily reflect the one fed from

the external device. In this case set the sampling rate as indicated by the *Measured Sampling Rate* within *Device Settings/General*.

Sync Source selects the clock to which the INT203 should sync to. For playback this is usually the INT203's *Internal* clock generator.

Operation Mode determines the stability of the system. Try other modes if clicks are encountered during playback.

Global Settings/Info Information about the driver version.

Device Settings/General The device settings should be pretty self-explanatory.

Device Settings/Firmware Loader Allows uploading new firmware to the INT203. Not used for normal operation.

Chapter 3

INT203 Technical Data

3.1 Digital Inputs

- (1) XLR connector
- (1) RCA connector
- all inputs accept professional or consumer standard, i.e. accept AES/EBU or S/PDIF signals.
- Maximum input wordlength: 24 bits
- The unit has to be power cycled in order for any primary input selection change to take effect (*IN1*/*IN2*)
- (2) Firewire connectors

3.2 Digital Outputs

- (1) XLR connector
- (1) RCA connector
- Professional Channel Status Data on the XLR and RCA outputs
- Single wire mode: stereo signal (channel 1/2) on all outputs (*OUT1*, *OUT2*), at all sampling rates
- (2) Firewire connectors

3.3 Synchronization

- Sampling Frequencies: 44.1, 48, 88.2, 96, 176.4 or 192 kHz
- Synchronized via an input signal (external sync source XLR or RCA) or the internal oscillator

- the computer is slaved to the INT203
- Extremely efficient Jitter attenuation down to subsonic frequencies

3.4 Power

Bus power from the Firewire connector or power from an external power supply, 8 . . . 12 V/0.2 A

3.5 Level Control

With the (optional) Remote Control unit the digital output gain can be controlled between 0 dB and -60 dB: From 0 dB to -20 dB in 0.5 dB steps and from -20 dB to -60 dB in 1 dB steps. In the 0 dB gain position the INT203 is fully bit transparent, i.e. it does not change the data in any way. In all other gain positions the output signal is properly dithered to a wordlength of 24 bits. All units are shipped with a 0 dB gain setting and non inverting polarity setting, i.e. when there is no remote control included the unit is in fully transparent mode.

3.6 Bit Transparency Check

Allows to check the bit transparency of the player program running on the computer. All standard sampling rates at 16 or 24 bit wordlenghts are supported.

3.7 Polarity Control

With the (optional) Remote Control unit the absolute polarity of the audio output signal can be controlled between 0° (not inverted) and 180° (inverted).

Appendix A

Contact

For any questions, suggestions etc. feel free to contact us at:

Weiss Engineering LTD.
Florastrasse 42
8610 Uster
Switzerland

Phone: +41 44 940 20 06
Fax: +41 44 940 22 14
Email: weiss@weiss.ch
Web: www.weiss-highend.com
www.weiss.ch
www.asiaweiss.com